MULTIPLE KERNEL LEARNING FOR SPEAKER VERIFICATION

C. Longworth and M.J.F. Gales

Engineering Department, University of Cambridge
Trumpington St, Cambridge, CB2 1PZ

ABSTRACT

Many speaker verification (SV) systems combine multiple classifiers using score-fusion to improve system performance. For SVM classifiers, an alternative strategy is to combine at the kernel level. This involves finding a suitable kernel weighting, known as Multiple Kernel Learning (MKL). Recently, an efficient maximum-margin scheme for MKL has been proposed. This work examines several refinements to this scheme for SV. The standard scheme has a known tendency towards sparse weightings, which may not be optimal for SV. A regularisation term is proposed, allowing the appropriate level of sparsity to be selected. Cross-speaker tying of kernel weights is also applied to improve robustness. Various combinations of dynamic kernels were evaluated, including derivative and parametric kernels based upon different model structures. The performance achieved on the NIST 2002 SRE when combining five kernels was 4.83% EER.

Index Terms— Speaker recognition, Dynamic kernels, Support Vector Machines, Classifier Combination

1. INTRODUCTION

Speaker Verification is a binary classification task in which the objective is to decide whether a given speech utterance was emitted by a specific claimed speaker. There has been considerable interest in applying Support Vector Machines (SVM) to this task. The SVM is a general purpose classifier that has been found to perform well on a wide range of tasks. Recent approaches such as [1] have shown gains by fusing the scores of multiple classifiers. For SVM-based systems, an alternative approach is to combine classifiers at the kernel level. This involves finding a suitable kernel weighting, known as Multiple Kernel Learning (MKL).

One approach to MKL is to perform a grid-search and select those weights that minimise the cross-validation error. However this approach is only practical for pairwise combination. An efficient, maximum-margin based scheme has recently been proposed in [2]. In this paper several refinements to maximum-margin MKL for speaker-verification are considered. The standard MKL scheme has a known tendency to yield sparse weightings. For a given set of kernels there is no guarantee that the level of sparsity is appropriate. A regularisation term is therefore proposed to allow the desired sparsity to be adjusted by the user. Unlike grid-search based MKL an optimal level of sparsity may be efficiently selected using cross-validation even when the number of kernels is high by tuning a single parameter. Cross-speaker tying of kernel weights is also considered. By defining the objective function over all enrolled speakers, a robust set of kernel weights may be obtained even when the available enrollment data per speaker is limited.

Maximum-Margin MKL is applied to combinations of two general classes of dynamic kernel, termed *parametric* and *derivative kernels*. These two forms of kernel are normally highly complementary although under certain conditions the associated features are known to be identical [3]. In [3] dynamic kernels were combined by concatenating feature spaces, weighting all kernels equally. This paper extends that work by by examining the case where kernels are individually weighted. Combination of dynamic kernels based upon different generative model structures is also evaluated. This paper is organised as follows. The next section describes dynamic kernels and introduces two categories of dynamic kernel, derivative and parametric kernels. In Section 3, Multiple Kernel Learning is discussed. In Section 4, experimental results on the NIST 2002 SRE dataset are presented. Finally conclusions are drawn.

2. DYNAMIC KERNELS

The Support Vector Machine is a binary discriminative classifier that has been successfully applied to a wide range of tasks. A useful property of the SVM is that it can be kernelised. During training and inference all references to data are in the form of inner products between training examples $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$. A *kernel function* $K(\boldsymbol{x}_i, \boldsymbol{x}_j)$ can be defined that implicitly calculates these inner products in some, potentially very high dimensional, feature space.

One disadvantage to using SVMs that that they can only classify data of some fixed dimensionality. By contrast, speech utterances are typically parameterised as variable length sequences of observations. One approach to overcoming this disadvantage is through the use of *dynamic kernels*. These typically make use of generative models and have the form

$$K(\boldsymbol{O}_i, \boldsymbol{O}_j; \boldsymbol{\lambda}) = < \boldsymbol{\phi}(\boldsymbol{O}_i; \boldsymbol{\lambda}), \boldsymbol{\phi}(\boldsymbol{O}_j; \boldsymbol{\lambda}) > \quad (1)$$

where $\boldsymbol{\lambda}$ is the set of parameters associated with a generative model and $\boldsymbol{\phi}(\boldsymbol{O}; \boldsymbol{\lambda})$ is a function that maps a speech utterance into a fixed dimensional feature space, known as a *score-space*. Many commonly used dynamic kernels can be placed into one of two classes, *parametric kernels* and *derivative kernels* [3], summarized below.

2.1. Parametric Kernels

Parametric kernels are a form of dynamic kernel where the features are the parameters $\boldsymbol{\lambda}$ associated with a generative model trained to represent an utterance $\boldsymbol{O} = \{\boldsymbol{o}_1, \ldots, \boldsymbol{o}_T\}$. Parametric score-spaces have the form

$$\boldsymbol{\phi}_{\boldsymbol{\lambda}}(\boldsymbol{O}; \boldsymbol{\lambda}) = \left[\hat{\boldsymbol{\lambda}}\right], \quad \hat{\boldsymbol{\lambda}} = \arg\max_{\boldsymbol{\lambda}} \{\log p(\boldsymbol{O}; \boldsymbol{\lambda})\} \quad (2)$$

A form of parametric kernel that is becoming increasingly widely used for speaker verification is the GMM-supervector kernel [4]. Here a GMM is used as the generative model and the feature space is formed by concatenating the means of an utterance-dependent GMM. As there is typically not enough observations per component to robustly estimate the model parameters, successive iterations of MAP adaptation, using the UBM as a prior, are typically used instead. For component m the MAP-adapted mean at iteration k is obtained using EM.

$$\boldsymbol{\mu}_m^{(k)} = \frac{\sum_{t=1}^{T} \gamma_m^{(k-1)}(t)\boldsymbol{o}_t + \tau\tilde{\boldsymbol{\mu}}_m}{\sum_{t=1}^{T} \gamma_m^{(k-1)}(t) + \tau} \quad (3)$$

where $\tilde{\boldsymbol{\mu}}_m$ are the UBM means associated with component m (which are also used as the initial parameters $\boldsymbol{\mu}_m^{(0)}$), $\gamma_m^{(k)}(t) = P(m|\boldsymbol{o}_t; \boldsymbol{\lambda}^{(k)})$, the posterior probability of component m at time t given observation $\boldsymbol{o}_t$ and $\boldsymbol{\lambda}^{(k)}$, and τ is the standard MAP adaptation constant that controls the influence of the prior on the final model

2.2. Derivative Kernels

Derivative kernels extract a fixed dimensional set of features from an utterance by calculating the derivatives of the log-likelihood of the utterances with respect to the parameters of a generative model. For a set of model parameters, $\boldsymbol{\lambda}$, the derivative feature-space generated from an utterance has the form

$$\phi_\nabla(\boldsymbol{O}; \boldsymbol{\lambda}) = \frac{1}{T}\left[\nabla_{\boldsymbol{\lambda}} \log p(\boldsymbol{O}; \boldsymbol{\lambda})\Big|_{\hat{\boldsymbol{\lambda}}}\right] \quad (4)$$

where $\hat{\boldsymbol{\lambda}}$ is the model parameter value at which the derivative is evaluated. An example of a derivative kernel that uses a GMM as a generative model is the Fisher kernel [5]. Here, derivatives with respect to the means of component m after k iterations of MAP adaptation are given by

$$\nabla_{\boldsymbol{\mu}_m} \log p(\boldsymbol{O}; \boldsymbol{\lambda})\Big|_{\boldsymbol{\lambda}^{(k)}} = \sum_{t=1}^{T} \gamma_m^{(k)}(t)\boldsymbol{\Sigma}_m^{-1}(\mathbf{o}_t - \boldsymbol{\mu}_m^{(k)}) \quad (5)$$

where $\gamma_m^{(k)}(t)$ is the posterior probability of component m generating $\mathbf{o}_t$ given $\boldsymbol{\lambda}^{(k)}$, and $\boldsymbol{\Sigma}_m$ is the covariance matrix associated with component m of the GMM.

2.3. Generative Model Structure

For dynamic kernels that incorporate a generative model, such as parametric or derivative kernels, an appropriate form of model must be selected. If a GMM is used, the number of Gaussian components must be chosen. This is a trade-off between improving the ability of the model to approximate the distribution over the acoustic space and ensuring that the model parameters can be robustly estimated with the available data. A suitable model size is typically chosen by selecting a value that reduces the error rate on some development dataset. As the trade-off is data-dependent this strategy may not be optimal.

If a suitable scheme for combining classifiers is available, then other strategies may be used. Rather than selecting a single form of model, a series of dynamic kernels can instead be defined, each based on different model structures. The associated classifiers can then be combined. Although this approach is more computationally expensive it has two advantages. Firstly, there is no need for prior knowledge about the task in order to select a suitable model size. Secondly, rather than making a single trade-off, the combined classifier can make use of features extracted from a range of different model structures, potentially leading to gains.

3. MULTIPLE KERNEL LEARNING

There has been considerable interest in combining multiple systems to improve performance. This is normally achieved by fusing the output scores of individual systems as in [1]. For SVM-based systems, an alternative approach is to combine at the kernel level. Given a set of K kernels, a combined kernel function may be defined as the weighted sum of the individual kernels.

$$\boldsymbol{k}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \sum_{k=1}^{K} \beta_k \boldsymbol{k}_k(\boldsymbol{x}_i, \boldsymbol{x}_j) \quad (6)$$

where $\beta_k \geq 0$ and $\sum_k \beta_k = 1$. Kernel function $\boldsymbol{k}_k(\boldsymbol{x}_i, \boldsymbol{x}_j)$, associated with kernel k, is defined by equation 1 for some function $\boldsymbol{\phi}_k(\boldsymbol{O}; \boldsymbol{\lambda})$. Learning a suitable set of weights is known as the Multiple Kernel Learning (MKL) problem. One approach to finding a suitable set of weights is to conduct a grid search over all possible weightings and select the weights that minimise the error. This MKL criterion is termed `minEER` when the Equal Error Rate metric is minimised. Unfortunately this approach is generally impractical for anything other than pairwise kernel combination.

An efficient approach to MKL was developed in [6] and extended in [2]. Here the kernel weights are incorporated into the standard SVM objective function. For a set of N utterances $\{\boldsymbol{O}_1, \ldots, \boldsymbol{O}_N\}$ each with associated label $y_i \in \{-1, 1\}$, the optimal set of weights are those that maximise the margin.

$$\begin{aligned}
\text{min} \quad & \frac{1}{2}\sum_{k=1}^{K}\frac{1}{\beta_k}||\boldsymbol{w}_k||_2^2 + C\sum_{i=1}^{N}\xi_i \\
\text{w.r.t} \quad & \boldsymbol{\beta}, \boldsymbol{w}_k, \boldsymbol{b}, \boldsymbol{\xi} \\
\text{s.t.} \quad & y_i\left(\sum_{k=1}^{K}\boldsymbol{w}_k^{\mathsf{T}}\boldsymbol{\phi}_k(\boldsymbol{O}_i; \boldsymbol{\lambda}) + b\right) \geq 1 - \xi_i \ \forall i \\
& \xi_i \geq 0 \ \forall i, \quad \beta_k \geq 0 \ \forall k, \quad \sum_{k=1}^{K}\beta_k = 1
\end{aligned} \quad (7)$$

where $\boldsymbol{w}_k$ are the primal SVM weights associated with kernel k and b, $\boldsymbol{\xi}$ and C are the standard SVM bias, slack vector and regularisation term. In this formulation $\boldsymbol{\beta}$ is subsumed into the definition of the primal weights and hence does not directly appear in the marginal constraint. There are a number of issues to address when applying this form of MKL directly to speaker verification.

Regularisation Term In equation 7, an l_1-norm constraint is applied to the kernel weights. A known consequence of this is to introduce a tendency towards sparse solutions [2]. For a given set of kernels, there is no guarantee that the level of sparsity will be optimal. One solution is to incorporate a regularisation term $\mathcal{R}$ into the objective function to allow the user to control the level of sparsity. A suitable form of regularisation is

$$\mathcal{R} = \varphi\sum_{k=1}^{K}(\beta_k - \frac{1}{K})^2 = \varphi\left(\sum_{k=1}^{K}\beta_k^2 - \frac{1}{K}\right) \quad (8)$$

due to the l_1-norm constraint on the kernel weights. Note that since the optimal solution is independent of any constant terms in the objective function $\mathcal{R} = \varphi\sum_{k=1}^{K}\beta_k^2$ may be used instead. Here φ is an empirically set constant. For large, positive values of φ the effect of this form of regularisation is to encourage a uniform set of weights. When φ is negative the solution will tend to be sparse and the objective function will perform kernel selection. Although an additional parameter has been introduced, note that an appropriate value for φ may be obtained through cross-validation even when the number of kernels is large.

Cross-Speaker Tying In most SVM-based speaker verification systems, a distinct set of SVM parameters is trained for each speaker. However, the amount of enrollment data available per speaker is typically limited. Additionally learning a set of speaker-dependent kernel weights may therefore lead to over-training. One way to obtain a more robust set of weights is to tie $\boldsymbol{\beta}$ over all enrolled speakers. This can be achieved by redefining the MKL objective function to sum over all speakers, while maintaining a separate set of marginal constraints for the enrollment data associated with each speaker.

Dynamic Range Normalisation The form of objective function given in (7) is biased towards those kernels for which the average magnitude of the associated feature vectors is greatest. Under a maximally non-committal kernel metric this corresponds to the kernels for which the associated score-space has the greatest dimensionality. It is therefore important that the kernel function includes some form of dynamic range normalisation. One option is Spherical Normalisation [7] where each feature vector is mapped onto the surface of a unit sphere. An alternative approach is to perform normalisation at the kernel level. Here, the features are simply duplicated for each kernel so all kernels have the same dimensionality.

The `maxMargin` MKL criterion used in this work is defined by the following objective function.

$$\begin{aligned} \text{min} \quad & \sum_{s=1}^{S}\left(\frac{1}{2}\sum_{k=1}^{K}\frac{1}{\beta_k}||\boldsymbol{w}_k^{(s)}||_2^2 + C\sum_{i=1}^{N}\xi_i^{(s)}\right) + \varphi\sum_{k=1}^{K}\beta_k^2 \quad (9) \\ \text{w.r.t} \quad & \boldsymbol{\beta}, \boldsymbol{w}_k, \boldsymbol{b}, \boldsymbol{\xi} \\ \text{s.t.} \quad & y_i^{(s)}\left(\sum_{k=1}^{K}\boldsymbol{w}_k^{(s)\mathrm{T}}\boldsymbol{\phi}_k(\boldsymbol{O}_i^{(s)};\boldsymbol{\lambda}) + b^{(s)}\right) \geq 1-\xi_i^{(s)} \ \forall i \ \forall s \\ & \xi_i^{(s)} \geq 0 \ \forall i \ \forall s, \quad \beta_k \geq 0 \ \forall k, \quad \sum_{k=1}^{K}\beta_k = 1 \end{aligned}$$

Where the speaker s ranges from $1 \ldots S$ and samples i range from $1 \ldots N^{(s)}$, $\boldsymbol{w}_k = \{\boldsymbol{w}_k^{(1)}, \ldots, \boldsymbol{w}_k^{(S)}\}$, $\boldsymbol{b} = \{b^{(1)}, \ldots, b^{(S)}\}$ and $\boldsymbol{\xi} = \{\xi^{(1)}, \ldots, \xi^{(s)}\}$. Equation 9 may be efficiently optimised by a similar approach to that used in [2]. Firstly, an equivalent constrained optimisation problem is defined.

$$\begin{aligned} \text{min}_{\,\boldsymbol{\beta}} \quad & \textstyle\sum_{s=1}^{S} J(s,\boldsymbol{\beta}) + \varphi\sum_{k=1}^{K}\beta_k^2 \\ \text{s.t.} \quad & \beta_k \geq 0 \ \forall k, \quad \textstyle\sum_{k=1}^{K}\beta_k = 1 \end{aligned} \quad (10)$$

Where $J(s,\boldsymbol{\beta})$ is the optimal value of the objective function associated with an SVM with kernel (6) and fixed kernel weights $\boldsymbol{\beta}$ after training on data associated with speaker s. A projected-gradient scheme can then be used to optimise (10). At each iteration $J(s,\boldsymbol{\beta})$ can be estimated using a standard efficient SVM implementation. An expression for the derivatives of $J(s,\boldsymbol{\beta})$ evaluated at $\boldsymbol{\beta}$ follows from the form in [2].

4. EXPERIMENTAL RESULTS

Various combinations of dynamic kernels were evaluated on the 2002 NIST SRE one-speaker detection task[8]. Each utterance was parameterised as sequences of 31-dimensional mel-PLP coefficients (15 static + 15 delta + delta energy) using a framerate of 10ms and a 30s window. To introduce additional robustness to noise, Cepstral Mean Subtraction was performed followed by Cepstral Feature Warping [9] using a three second window. Systems were primarily evaluated using the EER metric. To aid comparison with other work some minDCF scores are also quoted. The normalised DCF cost used in this paper takes the form DCF = $P_{\text{Miss}} + 9.9P_{\text{False Alarm}}$. minDCF is the minimum DCF score obtained a-posteriori by adjusting the decision threshold. Initially, gender-dependent UBMs were trained using EM for all SRE 2002 enrollment data. Each UBM consisted of a diagonal covariance GMM. For each enrolled speaker, a speaker-dependent GMM was constructed by MAP adapting the means of the appropriate gender-dependent UBM. Two iterations of static prior MAP were used with τ set at 25. These speaker-dependent models were used both as part of a LLR classifier and as the generative models for a derivative kernel. For this kernel the feature-space consisted of derivatives with respect to the GMM means. Parametric kernels were also used. Here utterance-dependent GMMs were obtained by adapting the appropriate UBM means using two iterations of MAP. For the parametric kernels τ was set at 5. Finally, for each utterance a parametric feature-vector was constructed by concatenating the GMM means. This setup was designed to avoid the conditions given in [3] under which derivative and parametric features are identical. During preliminary experiments, kernel-level normalisation, as described in Section 3, outperformed spherical normalisation and was used in these experiments to normalise the magnitude of the feature vectors. $\text{SV}M^{light}$[10] was used to train classifiers for each enrolled speaker. The SVM regularisation term C was left at the $\text{SV}M^{light}$ default. Imposter examples were obtained from the enrollment data associated with other speakers of the same gender. To reduce classifier bias each true utterance was duplicated until the two training sets were equal. For each kernel, a maximally non-committal distance metric was defined by normalising the global variance of each feature calculated over all speakers.

System	EER (%)	minDCF
GMM-LLR	12.10	0.4915
∇_{128}	8.62	0.3759
λ_{64}	9.55	0.3830
λ_{128}	8.61	0.3521
λ_{256}	8.58	0.3498
λ_{512}	8.83	0.3702
$\lambda_{128} + \nabla_{128}$	5.34	0.2300

Table 1. Comparison of baseline (equal-weight) kernel combination against derivative (∇), parametric (λ), and GMM-LLR systems

The performance of these initial systems is shown in Table 1. For 128-component models, derivative and parametric kernel performance was similar and both yielded significant gains compared to the GMM-LLR classifier. Initially, pairwise combination of 128-component derivative and parametric kernels was examined. An equally weighted combination, used in [3], was evaluated to provide a baseline. A 38% relative gain was observed compared to the parametric kernel alone. Gains were observed compared to [3] due to the improved parametric kernel obtained by tuning τ.

φ	Kernel Weights		EER (%)	minDCF
	∇_{128}	λ_{128}		
0	1.00	0.00	8.62	0.3759
0.008	0.80	0.20	7.11	0.3115
0.064	0.55	0.45	5.48	0.2378
∞	0.50	0.50	5.34	0.2300
`minEER`	0.39	0.61	4.93	0.2144

Table 2. Performance of `maxMargin` MKL combination as φ varies compared to optimal `minEER` weighting

Experiments were performed to identify whether individually weighting these kernels could yield gains compared to baseline combination. Initially, combination using a `minEER` criterion was evaluated. A line-search was performed and the kernel weights selected that gave the lowest EER. Although infeasible for larger number of kernels, this criterion forms a bound on the maximum gains obtainable using MKL. Next system combination was performed using the `maxMargin` criterion for MKL described in Section 3. $\boldsymbol{\beta}$ was tied over all speakers. Table 2 compares the performances obtained using `maxMargin` for a range of values of φ against the optimal `minEER`

weighting. When $\varphi = 0$ a sparse weighting is obtained that performs poorly compared to the baseline. This indicates that the default level of sparsity associated with MKL is not appropriate for this task. By increasing φ gains are observed. Note that for this configuration the objective function increases monotonically with φ and hence can not be used to select an appropriate regularisation factor. Best performance is achieved when $\varphi = \infty$. This case is equivalent to baseline combination. If a value for φ is chosen that minimises the EER, MKL is guaranteed to outperform or equal baseline combination. Unlike using the `minEER` criterion this is feasible for large numbers of kernels.

System	EER (%)	
	Equal-Weight	MKL
$\lambda_{64}+\lambda_{128}$	9.02	8.55
$\lambda_{128}+\lambda_{256}$	8.32	8.32
$\lambda_{256}+\lambda_{512}$	8.52	8.52
$\lambda_{64}+\lambda_{128}+\lambda_{256}+\lambda_{512}$	8.42	8.22
$\lambda_{128}+\nabla_{128}$	5.34	5.34
$\lambda_{64}+\lambda_{128}+\lambda_{256}+\lambda_{512}+\nabla_{128}$	5.22	4.83

Table 3. Comparison of equal-weight combination against `maxMargin` MKL for various combinations of kernels

The `maxMargin` MKL scheme was then applied to other combinations of kernels. In each case φ was adjusted a-posteriori to reduce the EER. Results are presented in Table 3. Combination of parametric kernels based upon different generative model structures was examined. Although no gains were observed for equal-weight combination of 64 and 128 component models, combination of 128 and 256 component models did yield small gains compared to the individual kernels. By comparison a 512-component system performed at 8.83% indicating that these gains were not simply due to the increased complexity of the combined classifier. For `maxMargin` MKL all pairwise combinations gave gains. These were cumulative when all four kernels were combined giving a 0.22% reduction in EER compared to equal-weight combination. Similar gains were observed in minDCF resulting in 0.3428 for four-way combination. The best overall performance was 4.83% (0.2079 minDCF) achieved when all kernels were combined. This represented a relative gain of 10% compared to the baseline and outperformed the optimal `minEER` pairwise combination by 0.10%. From the DET curve in Figure 1 it can be seen that this system performed best over the majority of the operating range. Additional gains may also be achievable by further combination with other forms of dynamic kernel such as the MLLR or CAT kernels, or by combination with dynamic kernels based upon other generative model structures.

5. CONCLUSIONS

This paper has looked at combining multiple dynamic kernels to improve performance of an SVM-based speaker verification system. One important question is how to learn an optimal kernel weighting, known as Multiple Kernel Learning. This paper examined a number of refinements to a recently proposed maximum-margin based scheme. The scheme has a known tendency towards sparse weightings, which may not be optimal for Speaker Verification. A regularisation term was proposed. This allows the user to tune the sparsity by adjusting a single parameter. Tying of kernel weights over all speakers was also applied to increase the robustness of the estimates. Combinations of dynamic kernels were evaluated on the NIST SRE02 task, including derivative and parametric kernels based around different generative model structures. The best performance

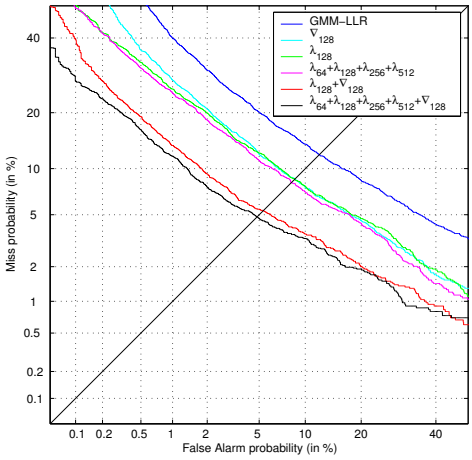

Fig. 1. DET graph comparing `maxMargin` MKL combination against individual systems

achieved was 4.83% EER obtained when all kernels were combined. This represented a 10% gain relative to an equal-weight pairwise baseline. The focus of this paper has been to give a general scheme for kernel combination. The range of kernels combined during evaluation was limited, using more diverse forms of kernel is expected to yield larger gains. Another area for future study is to contrast this scheme with standard score-fusion approaches.

6. REFERENCES

[1] W.M. Campbell, D. Sturim, W. Shen, D.A. Reynolds, and J. Navrtil, "The MIT-LL/IBM 2006 speaker recognition system: High-performance reduced-complexity recognition," in *Proc. ICASSP*, 2007.

[2] A. Rakotomamonjy, F. Bach, S. Canu, and Y. Grandvalet, "More efficiency in multiple kernel learning," in *Proc. ICMKL*, 2007.

[3] C. Longworth and M.J.F. Gales, "Derivative and parametric kernels for speaker verification," in *Proc. ICSLP*, 2007.

[4] W.M. Campbell, D. Sturim, D.A. Reynolds, and A. Solomonoff, "SVM based speaker verification using a GMM supervector kernel and NAP variability compensation," in *Proc. ICASSP*, 2006.

[5] T. Jaakkola and D. Haussler, "Exploiting generative models in discriminative classifiers," in *NIPS*, 1999.

[6] S. Sonnenburg, G. Rätsch, and C. Schäfer, "A general and efficient multiple kernel learning algorithm," *Advances in Neural Information Processing Systems*, 2005.

[7] V. Wan and S. Renals, "Speaker verification using sequence discriminant support vector machines," *IEEE Transactions Speech and Audio Processing*, 2004.

[8] A. Martin, "The NIST year 2002 speaker recognition evaluation plan," 2002, Available from http://www.nist.gov/speech/tests/spk/2002/doc.

[9] J. Pelecanos and S. Sridharan, "Feature warping for robust speaker verification," in *Proc. ISCA Workshop on Speaker Recognition - 2001: A Speaker Oddyssey*, 2001.

[10] T. Joachims, "Making large-scale SVM learning practical," in *Advances in Kernel Methods - Support Vector Learning*, B. Scholkopf, C. Burges and A. Smola, Ed. MIT Press, 1999.